**CHAPTER 1**

**1.1 INTRODUCTION**

The impact of computers on our lives today is probably much more than we are actually known to. Getting good information and transforming it quickly into products than consumers want to buy is the essential key to staying in business and this all is done nowadays using Computers and Application Software. In order to maintain competitiveness in today’s global market, industry requires systems that are capable of quickly responding to change while maintaining stable and efficient Operation. Increasingly, the industrial control system is viewed as being central to achieving this goal.

Automated software engineering applies computation to software engineering activities. The goal is to partially or fully automate these activities, thereby significantly increasing both quality and productivity. This includes the study of techniques for constructing, understanding, adapting and modelling both software artefacts and processes. Automatic and collaborative systems are both important areas of automated software engineering, as are computational models of human software engineering activities. Knowledge representations and artificial intelligence techniques applicable in this field are of particular interest, as are formal techniques that support or provide theoretical foundations.

Automated software engineering approaches have been applied in many areas of software engineering. These include requirements definition, specification, architecture, design and synthesis, implementation, modelling, testing and quality assurance, verification and validation, maintenance and evolution, configuration management, deployment, reengineering, reuse and visualisation. Automated software engineering techniques have also been used in a wide range of domains and application areas including industrial software, embedded and real-time systems, aerospace, automotive and medical systems, Web-based systems and computer games.

**1.2 LITERATURE SURVEY**

Dr. N. Jesse(2010), Business Process Automation for Universities Enabling Effective and Efficient University Management, [2010 International Conference on Management and Service Science](http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=5575324), IEEE, 1-4

**I.**

Though universities have been regarded as an unusual institution which is open to criticism and to express voices and intellectual assets and run by committees and trusties yet another perspective has emerged where it has been regarded as a strategic driving force for a knowledge based society. Implementing new technologies and trends, universities should be ready to transform into a digitalized institutions by using the full potential of Information Technology.

**II.**

Powerful universities use powerful technologies to ensure the digitalization of the whole system and in doing so they need to keep pace with the technological progress on the hardware and network layer which does not seem of that challenge. Only few universities have developed transparent IT strategies for change management, restructuring, re-strategizing, quality efforts etc. [3]**.** In times of the dominance of the web and evolving Web 2.0 applications, the global trend is to make central business processes accessible anytime, anywhere. Experimentation has always been there using eLearning platforms, electronic workspaces or podcasts, but these should be embedded in a way to know the real potential of automation.

**III.**

Taking Germany as an example it is seen that IT support has a strong focus on student enrolment, accounting, aspects of human management, examination management and course management. Although these applications are interconnected but are inflexible in terms of adaptation and customization. Effective and efficient IT support of students, teachers, and researchers is essential if a university wants to remain competitive. The focus should be on electronic workflow solutions exploiting modern software concepts, technologies and applications. The focal point of the system must be the flow of documents, data and responsibilities. Electronic workflow systems contribute to the “automation of an entire or part of a business process where tasks, procedural steps, organizations or people involved, required input and output information needed for each step”.

**1.3 EXISTING METHOD**

College automation system is the software which gathers the basic information of students. It is becoming a very essential component in education in this modern age. With the help of College Automation System, we can gather all the useful information needed to the management in few clicks. Our college Automation System consists of different modules such as student, faculty, admin etc. And manage the working of these different modules. The interconnectivity among modules reduces the time to perform different operational task, but there are some problems still exists with our system which has been highlighted below.

In Existing System for managing various modules such as Student module, Administrative module and Exam cell etc. takes lot of paper work as well as time. For instance our examination cell use a lot of paper work for semester examination notice, time table and siting arrangements. In Students module at present feedback system is a major issue for a lot of students. Here also we use a lot of paperwork at the end of a semester. In Administrative module Exam Form, Concession Form, Accounts and updating details, Profile views, Fees details, ID card generation all are very difficult to manage using manual processes and it takes lot of time and paper work.

**CHAPTER 2**

**PROPOSED METHOD**

In the University Automation System (U.A.S.) we are providing role based access rights to different users, that are students, teacher and administrator. In our system the students and teachers can view all the Notices such as examination notice, marks notice, announcement notice, training and placement notice, schedule notice etc. which are published by the respective departments. To improve thyself feedback is very important. Keeping this in mind, we have a section in our system dedicated to feedback. With the help of this, students can give feedback to their respective teachers which will help in increasing the quality of education in a university. In U.A.S. (University Automation System), students and teacher can access their class time table, anywhere and on the go without being a hassle to go to a particular place to see the time table. Students can access all their academic details with the help of an interactive graph. The system is very user friendly, it is made in such a way that users will have a hassle free and interactive user experience.

**CHAPTER 3**

**Introduction Theory**

In this chapter we will discuss about the Java as a programming language and how it is used to develop various graphical user interface systems using Java swing components and JDBC which allows us to connect a database with desktop software.

We will build a software on the lines of student management system which would include not only student account but even teacher accounts for various previous lacking exchanges between them.

**3.1 Design Cycle**

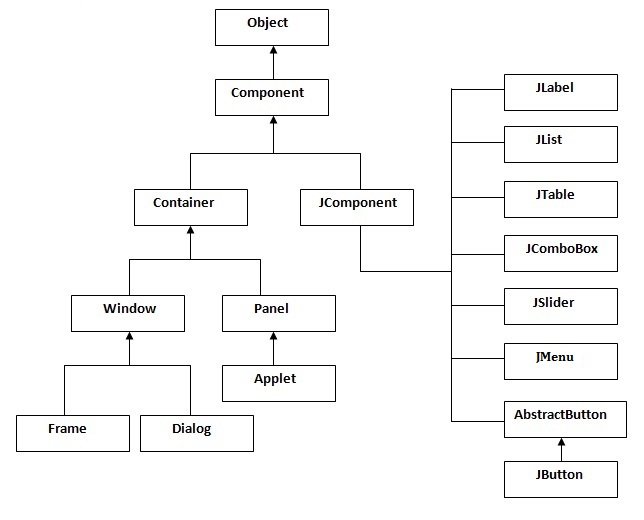
**3.1.1 Flow Chart**

**3.2 Graphical User Interface using Java Swings**

Java Swing is a part of Java Foundation Classes (JFC) that is used to create window-based applications. It is built on the top of AWT (Abstract Windowing Toolkit) API and entirely written in java.

Unlike AWT, Java Swing provides platform-independent and lightweight components.

The javax.swing package provides classes for java swing API such as JButton, JTextField, JTextArea, JRadioButton, JCheckbox, JMenu, JColorChooser etc.

**JFrame** –frame is an instance of JFrame. Frame is a window where we can have title, border, menu, buttons, text fields and several other components. Any swing application must have a frame in it to store other swing components.

**JPanel** – A panel is an instance of JPanel. A frame can have more than one panels and each panel can have several components. You can also call them parts of Frame. Panels are useful for grouping components and placing them to appropriate locations in a frame.

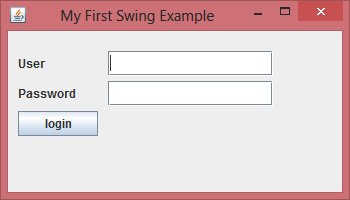
**JLabel** – A label is an instance of JLabel class. A label is unselectable text and images. If you want to display a string or an image on a frame, you can do so by using labels. In the above example we wanted to display texts “User” & “Password” just before the text fields, we did this by creating and adding labels to the appropriate positions.

**JTextField** – Used for capturing user inputs, these are the text boxes where user enters the data.

**JPasswordField** – Similar to text fields but the entered data gets hidden and displayed as dots on GUI.

**JButton** – A button is an instance of JButton class. In the above example we have a button “Login”.

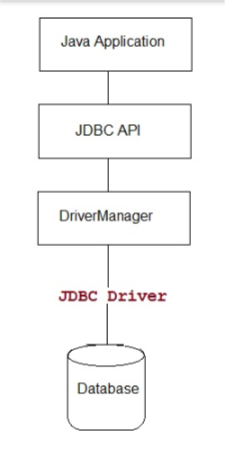
A simple swing example



This is a very basic example of the how the system will interact with the user using the JFrame as the outer container with the JTextfield and JLabel as the elements of the JPanel inside it with a JButton to use it to move to the next frame after validating the credentials of the user.

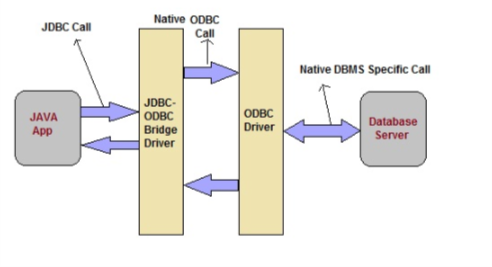
**3.3 JDBC (Java Database Connectivity) connection to the GUI**

**Java Database Connectivity(JDBC)** is an **Application Programming Interface(API)** used to connect Java application with Database. JDBC is used to interact with various type of Database such as Oracle, MS Access, My SQL and SQL Server. JDBC can also be defined as the platform-independent interface between a relational database and Java programming. It allows java program to execute SQL statement and retrieve result from database.

****

**3.3.1** There are 4 different types of drivers present in JDBC.

* We are using **Type-1 Driver** or **JDBC-ODBC bridge**

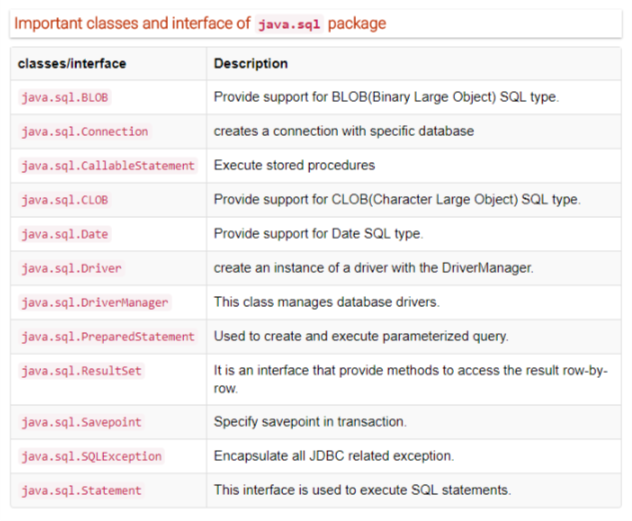


**Advantage**

* Easy to use
* Allow easy connectivity to all database supported by the ODBC Driver.

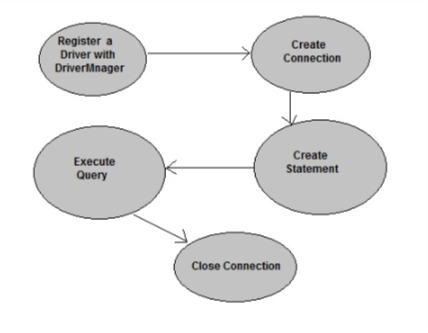
**Disadvantage**

* Slow execution time
* Dependent on ODBC Driver.
* Uses Java Native Interface(JNI) to make ODBC call.

**3.3.2 Some Important classes and interfaces from java.sql**

**3.3.3 The following 5 steps are the basic steps involve in connecting a Java application with Database using JDBC.**

1. Register the Driver
2. Create a Connection
3. Create SQL Statement
4. Execute SQL Statement
5. Closing the connection



**3.3.4** Steps for connecting jdbc with GUI

* **Establishing a connection**

**Connection** establishing a connection between the GUI and the database using the connection class statement.

* **Creating a jdbc statement object**

**Statement:** Execute simple sql queries without parameters.  
      Statement createStatement()  
    Creates an SQL Statement object.

* **Executing a SQL statement with the Statement object, and returning a jdbc ResultSet.**

**ResultSet** provides access to a table of data generated by executing a Statement. The table rows are retrieved in sequence. A ResultSet maintains a cursor pointing to its current row of data. The next() method is used to successively step through the rows of the tabular results.

**3.4 Creating User friendly experience**

To the benefit of software and applications everywhere, UX has become an increasingly important step in the Software Development lifecycle. Even though UX is a slightly surreptitious layer of design, it’s no less important. It, by the nature of its name, defines the experience of the users. That experience determines whether or not they want to come back for more, or run screaming in the other direction.

UX is something anyone can do. The problem is not everyone can do it well. We can undertake the visual design for a website, but we guarantee you that it will look dreadful and unprofessional.

If you learn nothing else about UX we ask you to remember these two things.

1. Know the user.
2. You are not the user (in most cases).

If you don’t know your user base, their habits and tendencies, and why they do or do not perform certain actions, then you can’t be expected to design a good experience for them. Even if you are very similar to your users, remember that you are only one person and that does not define the qualities of your user base as a whole. Get to know your users and design the experience with them in mind. The role of a UXer is to have that knowledge, constantly expand it, and then design with those users in mind, crafting amazing experiences that will delight and fulfill them.

Let’s look at all the possible steps a UXer might take in defining the UX of a product. These steps create the ideal process. These steps aren’t always possible to complete in the real world, but we need to cover all of them so that you’re aware when you can leave out certain steps and why. Sometimes you don’t necessarily leave out a step, but it is melded into another step, or replaced using a combination of experience, knowledge, and intuition.

**3.4.1 Requirements**

Requirements gathering is one of the first steps in designing the UX. During this stage you need to ask a lot of questions. Many questions may not be able to be answered right away but note those and be persistent. There are several types of requirements:

1. **Business Requirements**: The goals and needs of other parts of your company or what’s necessary to monetize the product. Unfortunately, this often trumps some things you may want to do. It’s a necessary evil if your product is anything beyond a project for pure enjoyment.
2. **Design Requirements**: Sometimes there may be special design considerations or needs that must be met.
3. **Technology Requirements**: There could be a specific technology need (platform, language, etc.) you need to consider in the design. What are your limitations?
4. **User Requirements**: Who is this product for? Who is the main audience? Is there a fringe audience and, if so, who is it? Does it cover your entire user base or support a subset?

**CHAPTER 4**

**Future work to be done**

* Teacher – Student feedback system, which will enable students to give their respective feedbacks to their teachers directly in real time.
* Timetable portal, which will enable students to view the time table of their respective classes. If further implemented using B.P.A (business process automation) or R.P.A(robotics process automation), then the timetable for whole department could be easily generated without any human interference with the help of Artificial Intelligence (A.I.)
* Students can also use the U.A.S (University Automation System) in their mobile phones and tablets through an Android and iOS application.
* Students can also chat with the faculties to clear doubts and ask questions using the real time chat system.
* Students can get counselling with the help of A.I. Chat bots and also can chat with the counsellors present in the university.
* It will also show the graph of Cumulative Grade Point Average (C.G.P.A.) and Semester Grade Point Average (S.G.P.A.) with respect to semester. And also use the U.A.S (University Automated System) to view the marks and answer sheets.